20050231-925TS6S9

Architecture 100

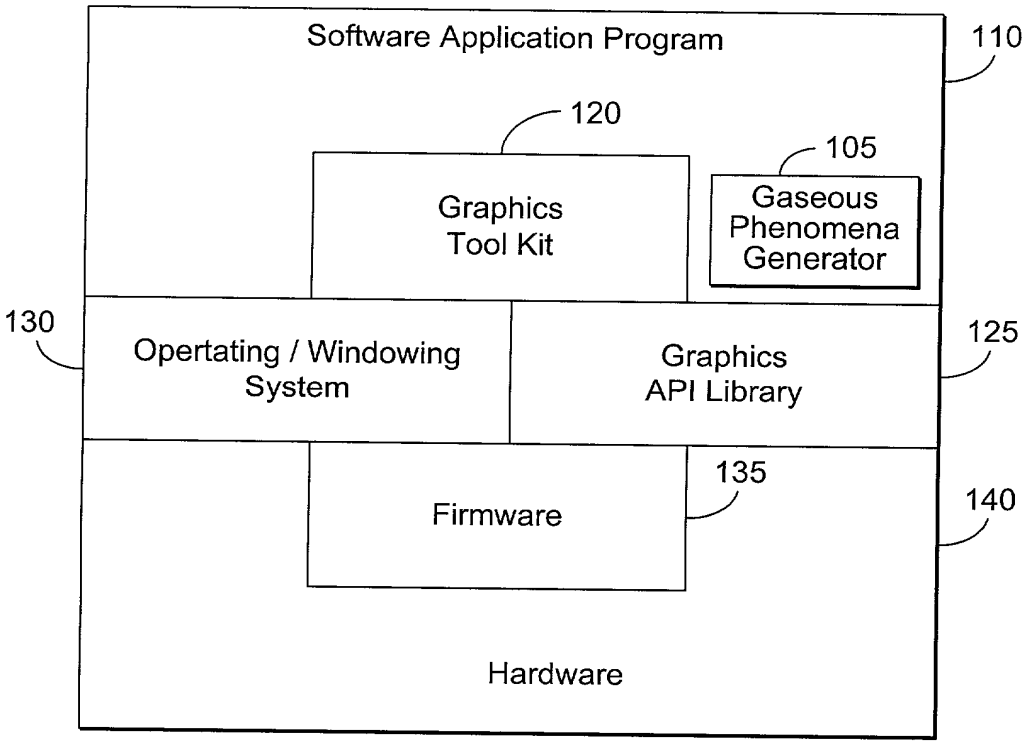


FIG. 1

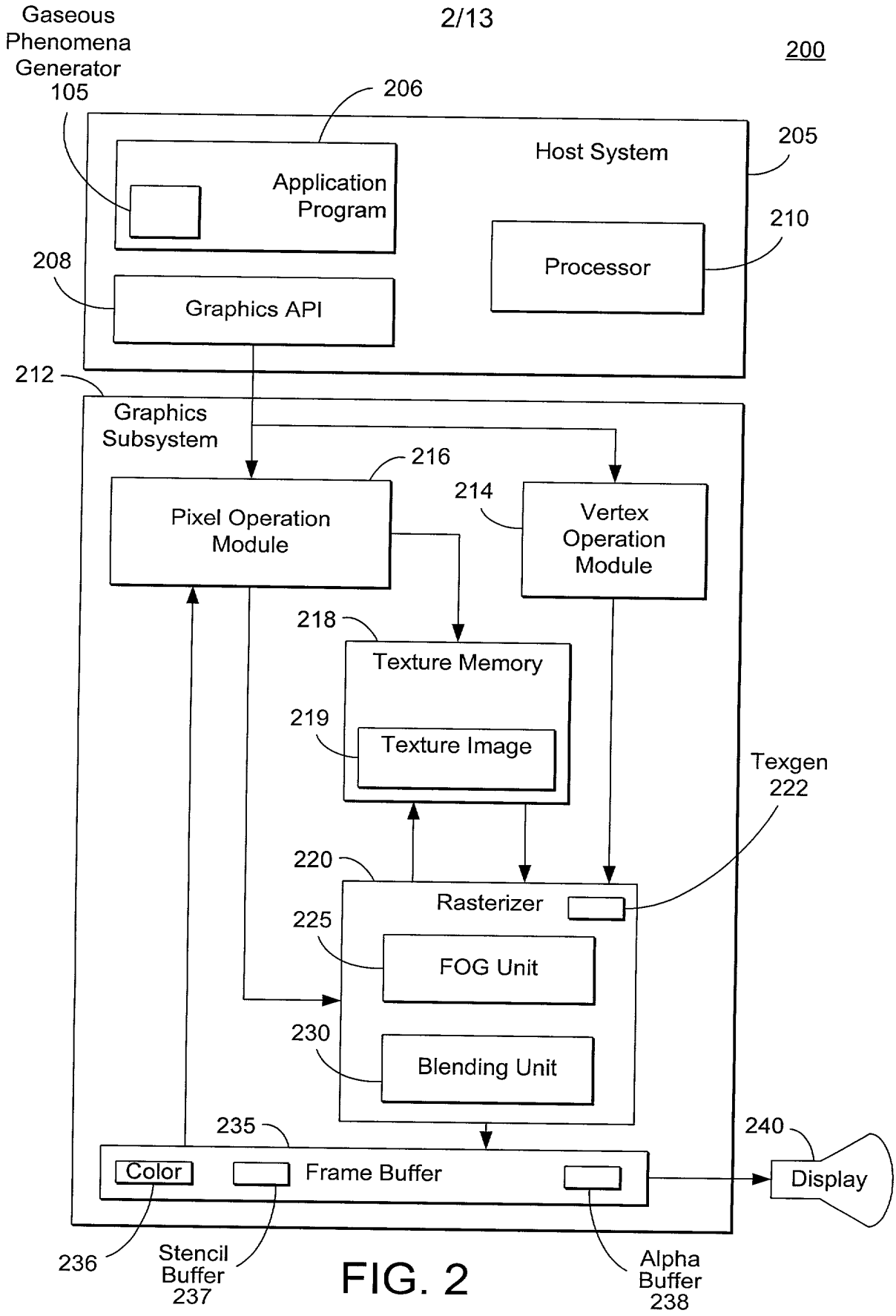


FIG. 2

20050225-5256-9940

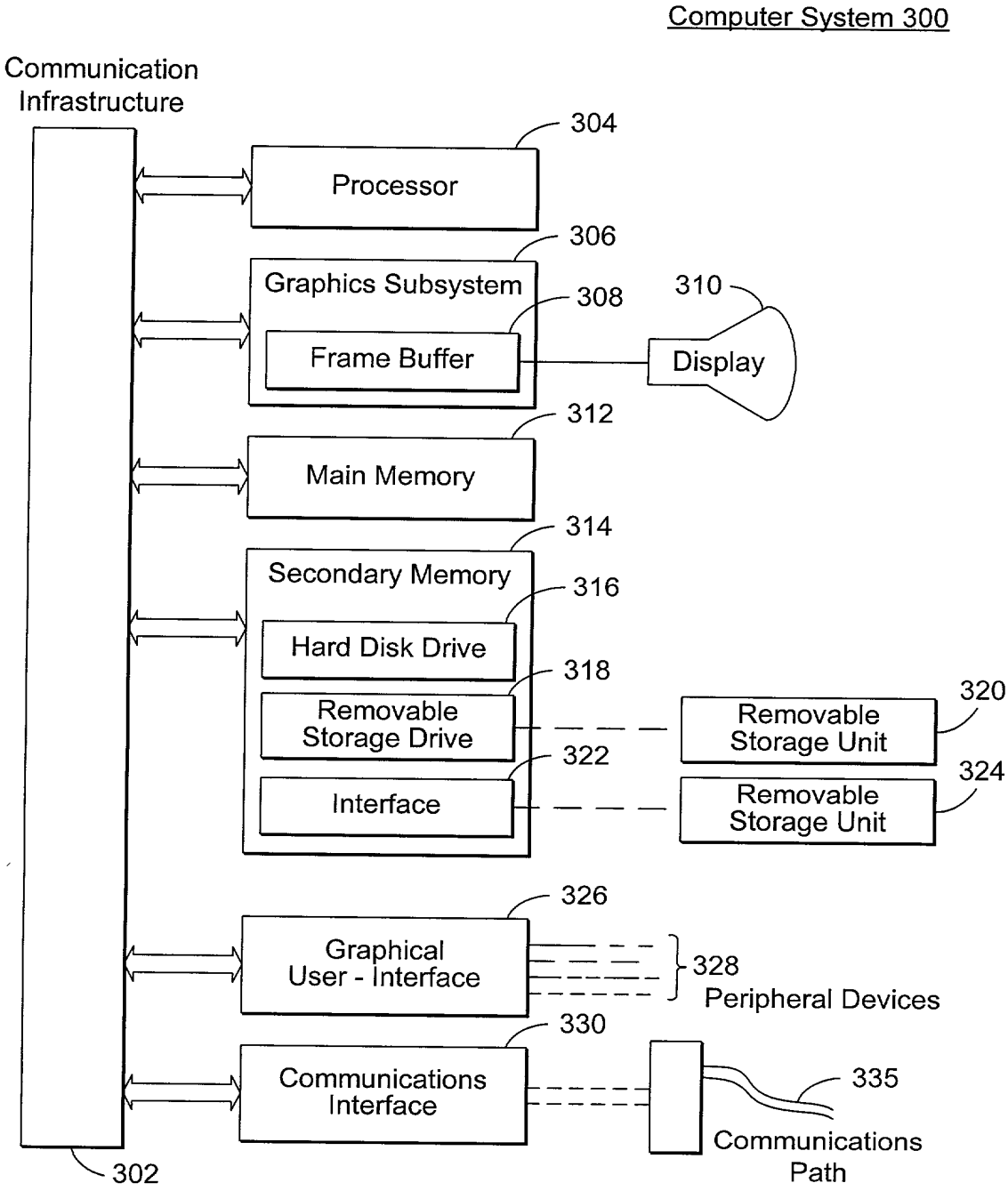


FIG. 3

20050225-0225050

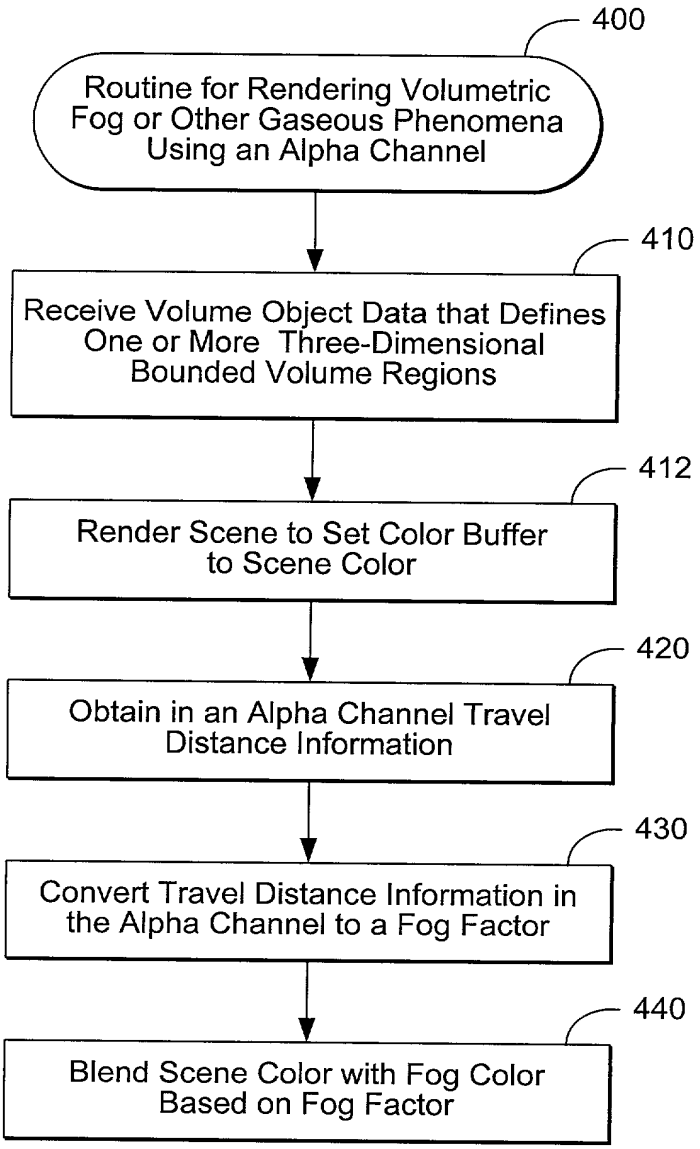


FIG. 4

09945526.020502

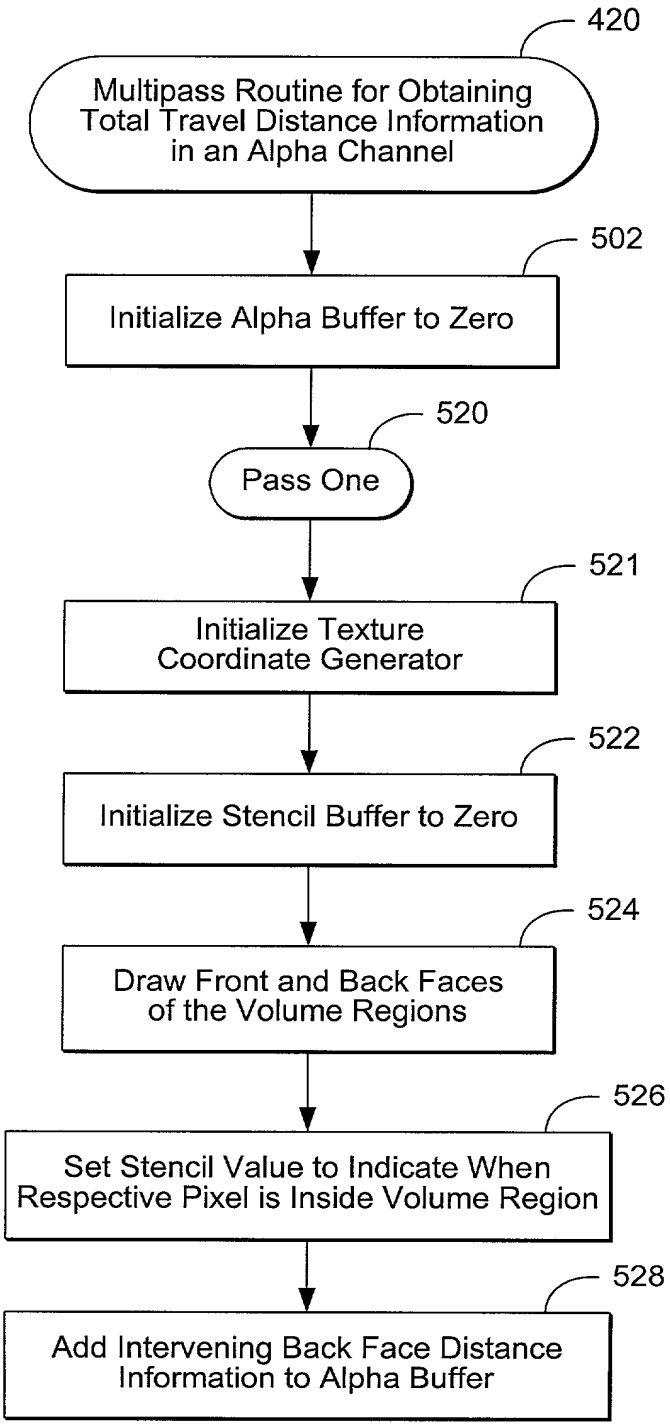

420
Multipass Routine for Obtaining Total Travel Distance Information in an Alpha Channel
502
Initialize Alpha Buffer to Zero
520
Pass One
521
Initialize Texture Coordinate Generator
522
Initialize Stencil Buffer to Zero
524
Draw Front and Back Faces of the Volume Regions
526
Set Stencil Value to Indicate When Respective Pixel is Inside Volume Region
528
Add Intervening Back Face Distance Information to Alpha Buffer

FIG. 5A

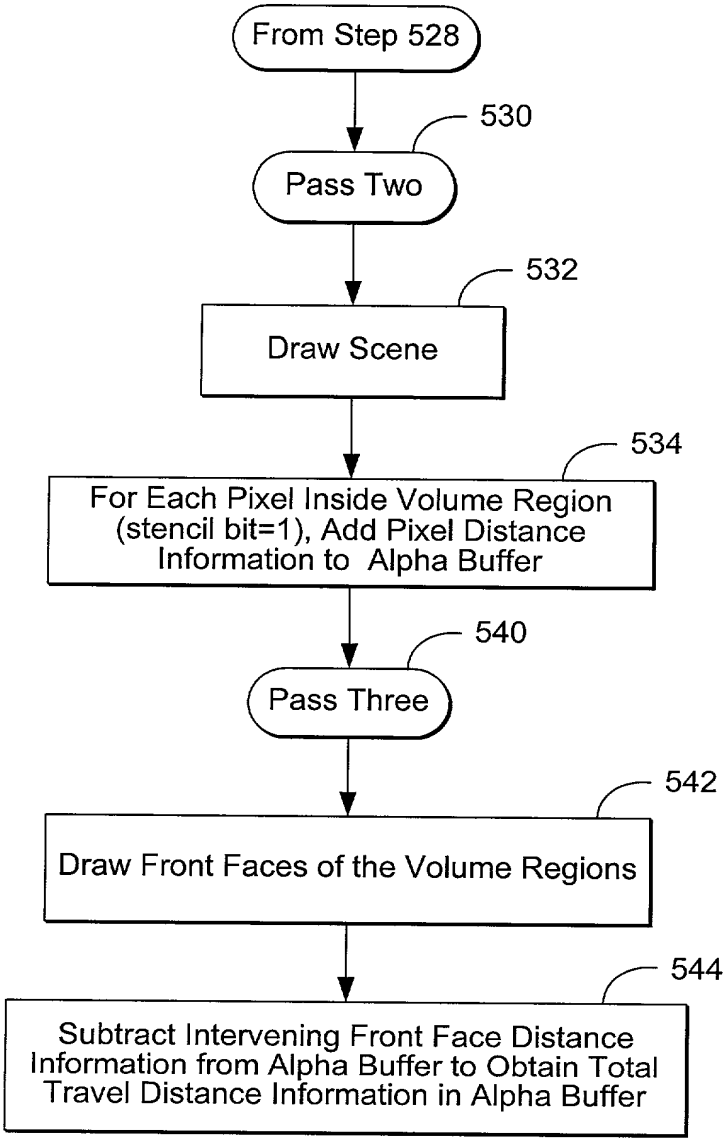


FIG. 5B

20050225-020502 Rejected

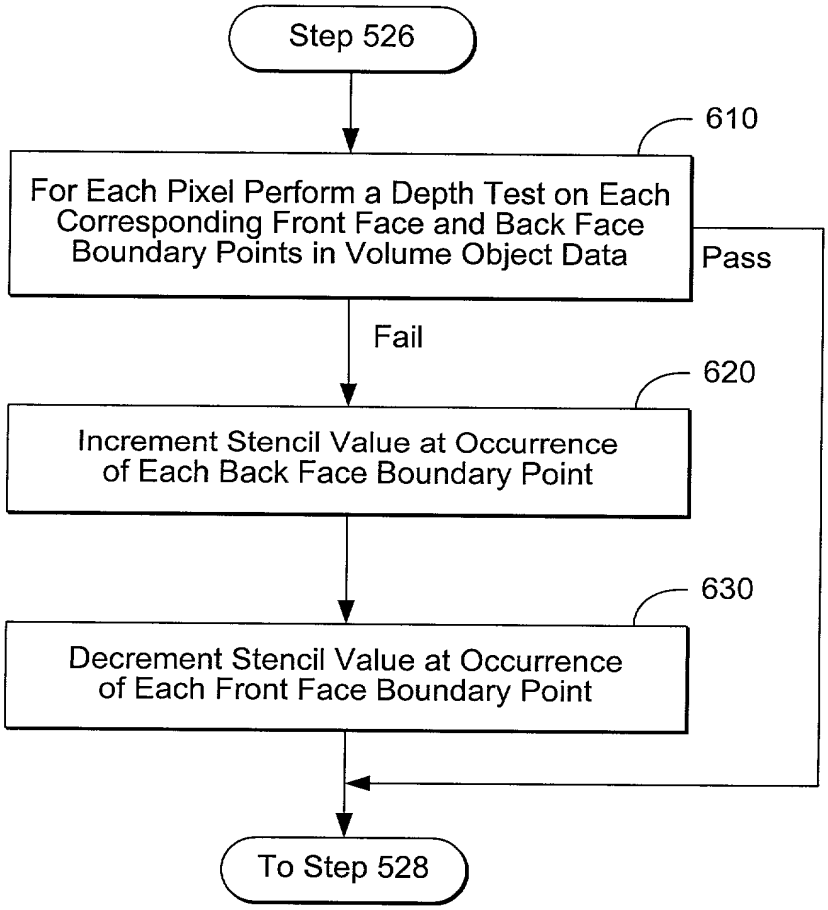


FIG. 6

9994525-020502

20030203-3257660

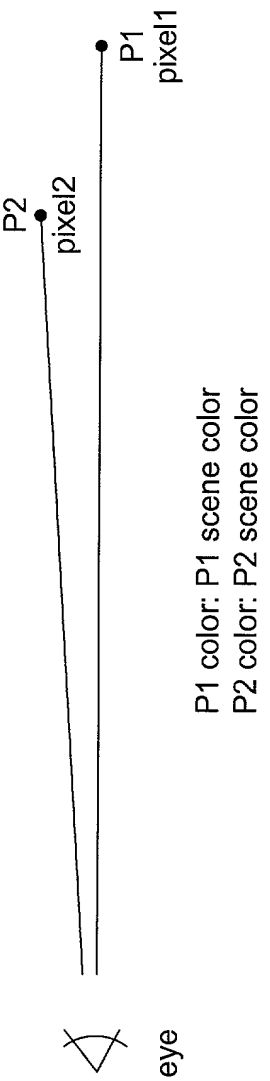


FIG. 7A

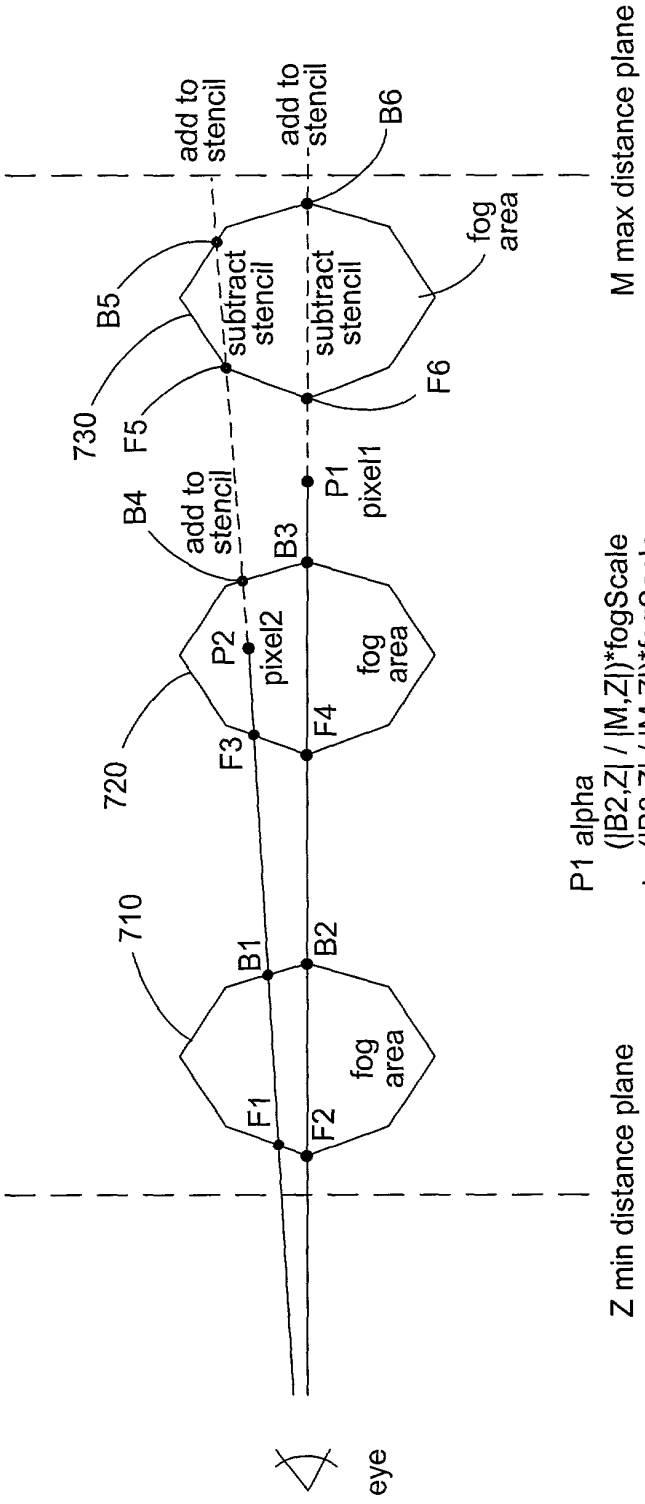
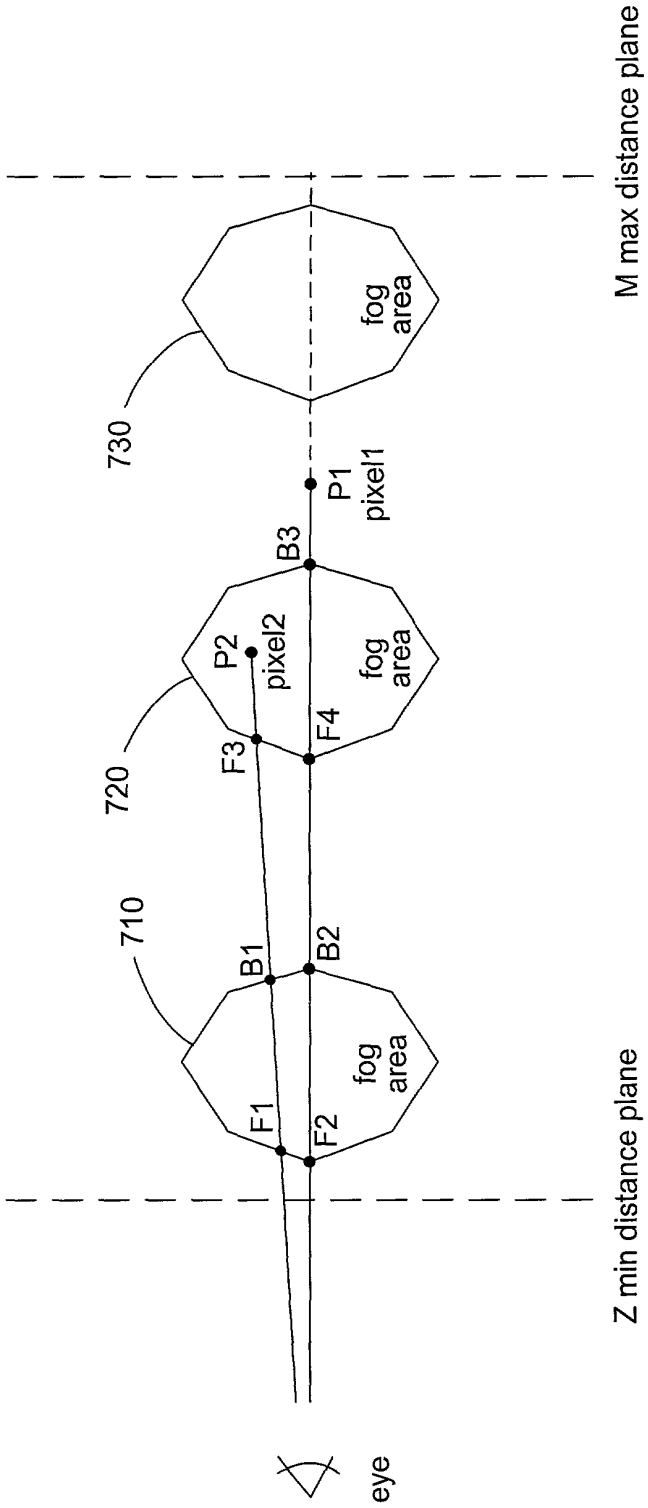


FIG. 7B

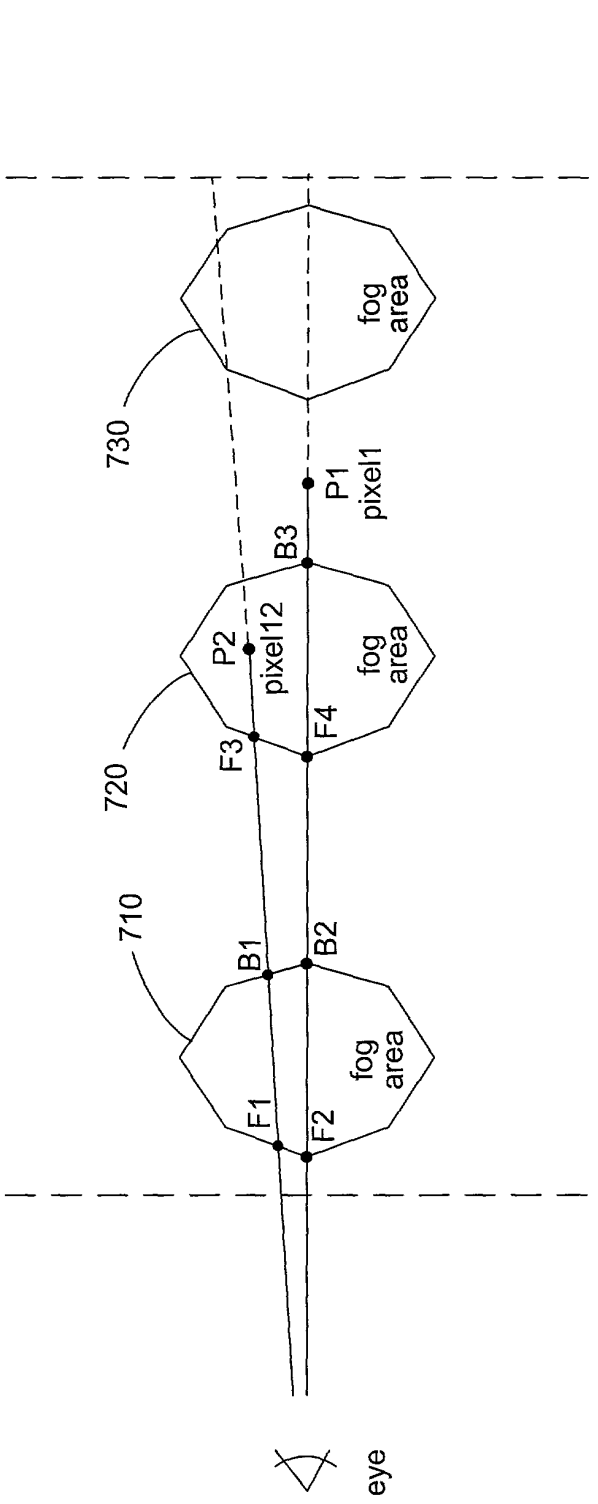
20020 - 9251660



P1 alpha : not changed, stencil 0
= ((|B2,Z| + |B3,Z|) / |M,Z|)*fogScale

P2 alpha : changed, due to stencil 1
P2 alpha
+ (|P2,Z| / |M,Z|)*fogScale
= ((|B1,Z| + |P2,Z|) / |M,Z|)*fogScale

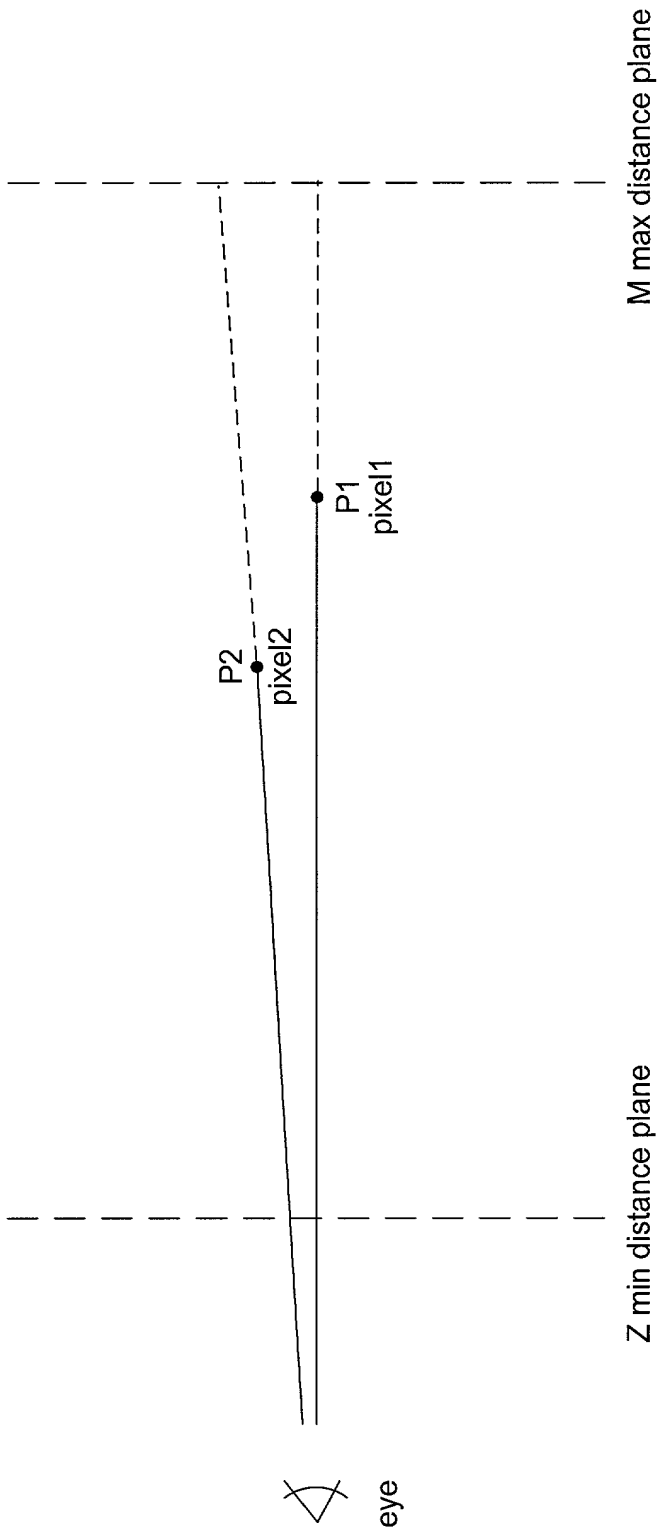
FIG. 7C



P1 alpha:
P1 alpha
- (|F4,Z| / |M,Z|)*fogScale
- (|F2,Z| / |M,Z|)*fogScale
= (((|F2,B2| + |F4,B3|) / |M,Z|)*fogScale

P2 alpha:
P2 alpha
- (|F1,Z| / |M,Z|)*fogScale
- (|F3,Z| / |M,Z|)*fogScale
= (((|F1,B1| + |F3,P2|) / |M,Z|)*fogScale

FIG. 7D



Fog Factor { P1 alpha: P1 alpha *(fogDensity/fogScale)*|M,Z|
P2 alpha: P2 alpha *(fogDensity/fogScale)*|M,Z|
(linear fog)

or

Fog Factor { P1 alpha: pixelmap[(P1 alpha/fogScale)*|M,Z|]
P2 alpha: pixelmap[(P2 alpha/fogScale)*|M,Z|]
(exp or exp2 fog)

FIG. 7E

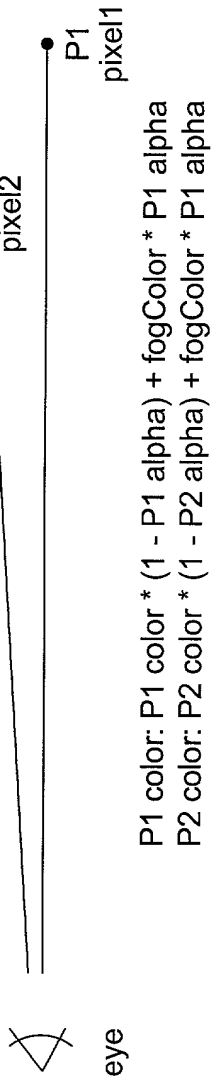


FIG. 7F

20070720-9257660